

## Тема: Виртуални функции, множествено наследяване:

1. Към класа CAddress от упражнение 4 да се добави глобална приятелска функция (оператор) за четене на данните за адреса по указан като параметър входен поток:  
`friend ostream& operator>>(ostream& fromStream, CAddress& addr)`
2. Към класа CPerson от упражнение 4 да се добави глобална приятелска функция (оператор) за четене на данните за личността по указан като параметър входен поток.
3. Към класа CStudent от упражнение 4 да се добави глобална приятелска функция (оператор) за четене на данните за студента по указан като параметър входен поток.
4. Да се състави абстрактен клас CGSM, предназначен за наследяване от класа CStudent от упражнение 4. Частни член променливи и конструктори:
  - Модел - низ
  - Тарифен план – число
  - Конструктори – подразбиращ се и експлицитенПублични напълно виртуални функции:  
`virtual ostream& OutputGSM(ostream& toStream ) const = 0;`  
`virtual string getGSMNumber() const = 0;`  
`virtual istream& InputGSM(istream& fromStream ) = 0;`
  - Оператор за четене от входен поток : `operator>>`
5. Класа CStudent от упражнение 4 да се направи наследник на втори клас CGSM. Да се добави член променлива номер на телефон - низ.
6. Към класа CStudent от упражнение 4 да се добави сравнител на студенти, който не е глобална приятелска функция. Сравнението да се имплементира като се сравнят по тарифен план, напр.:  
`bool operator>(const CStudent& studR)const`
7. Да се състави подходяща главна функция за проверка на създадените функции и оператори.
8. В главната функция да се добави четене на данни за масив от студенти от файлов поток, например:  
`fstream iFile("Tema05.txt", ios_base::in);`